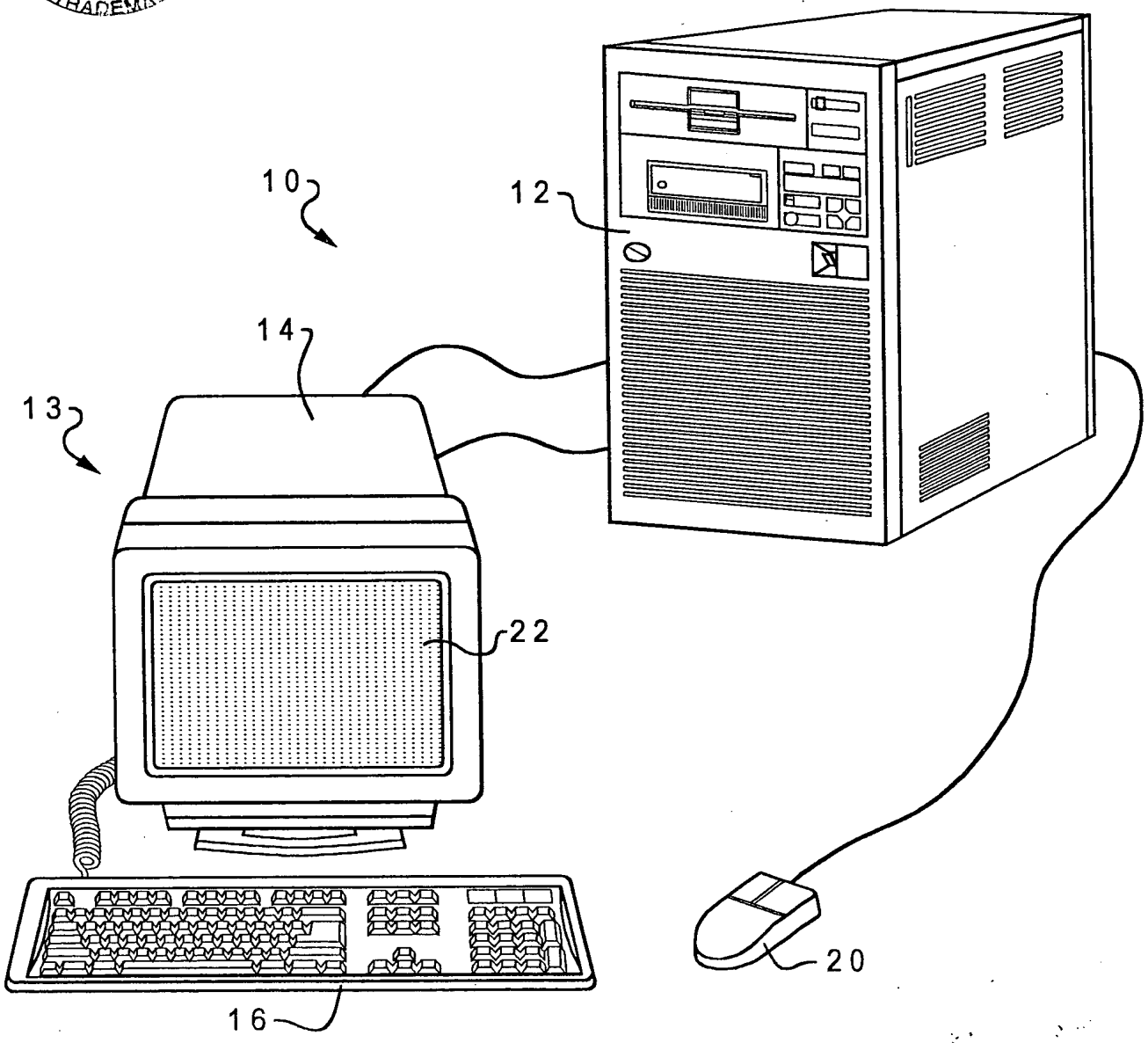


Fig. 1

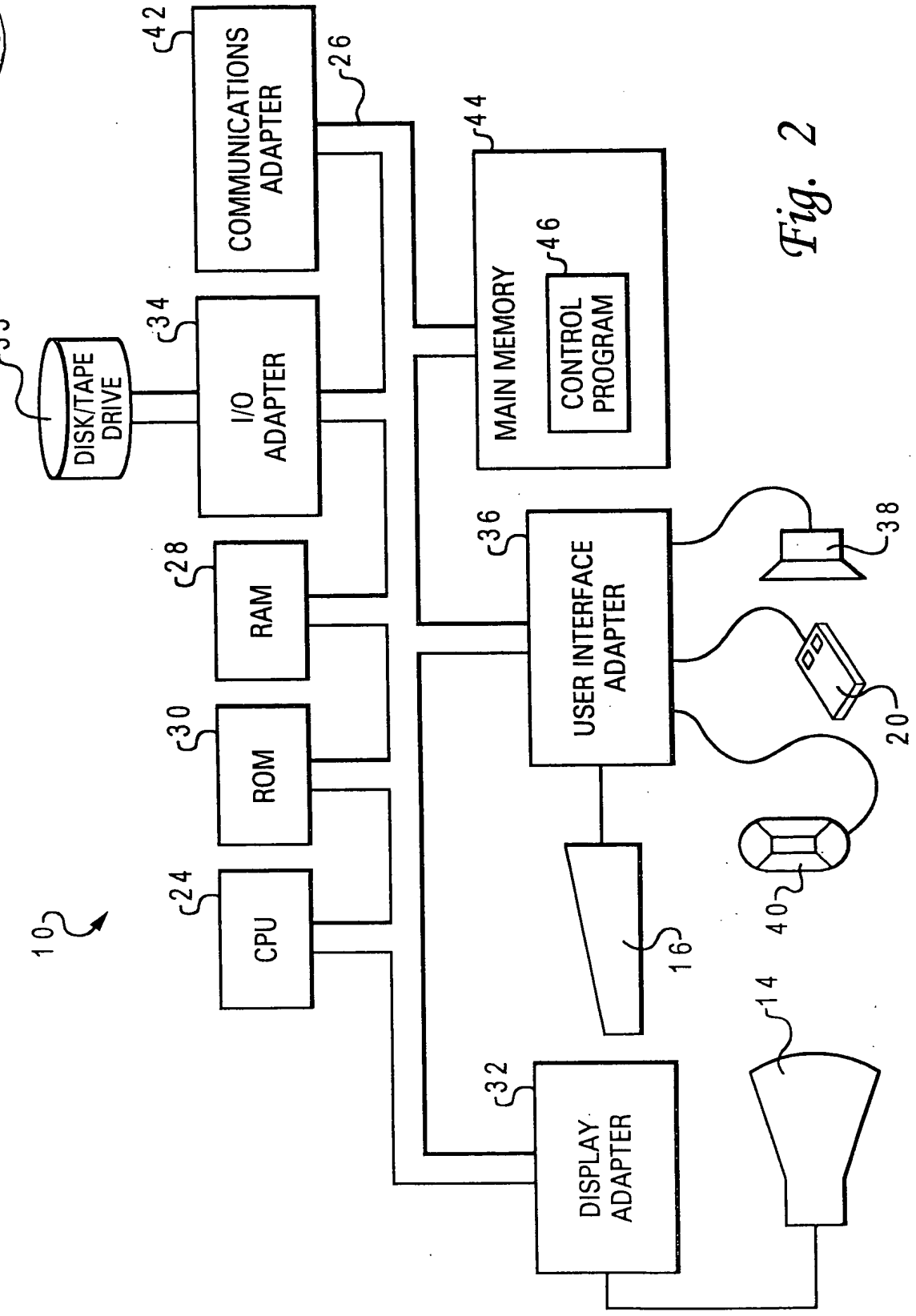


Fig. 2

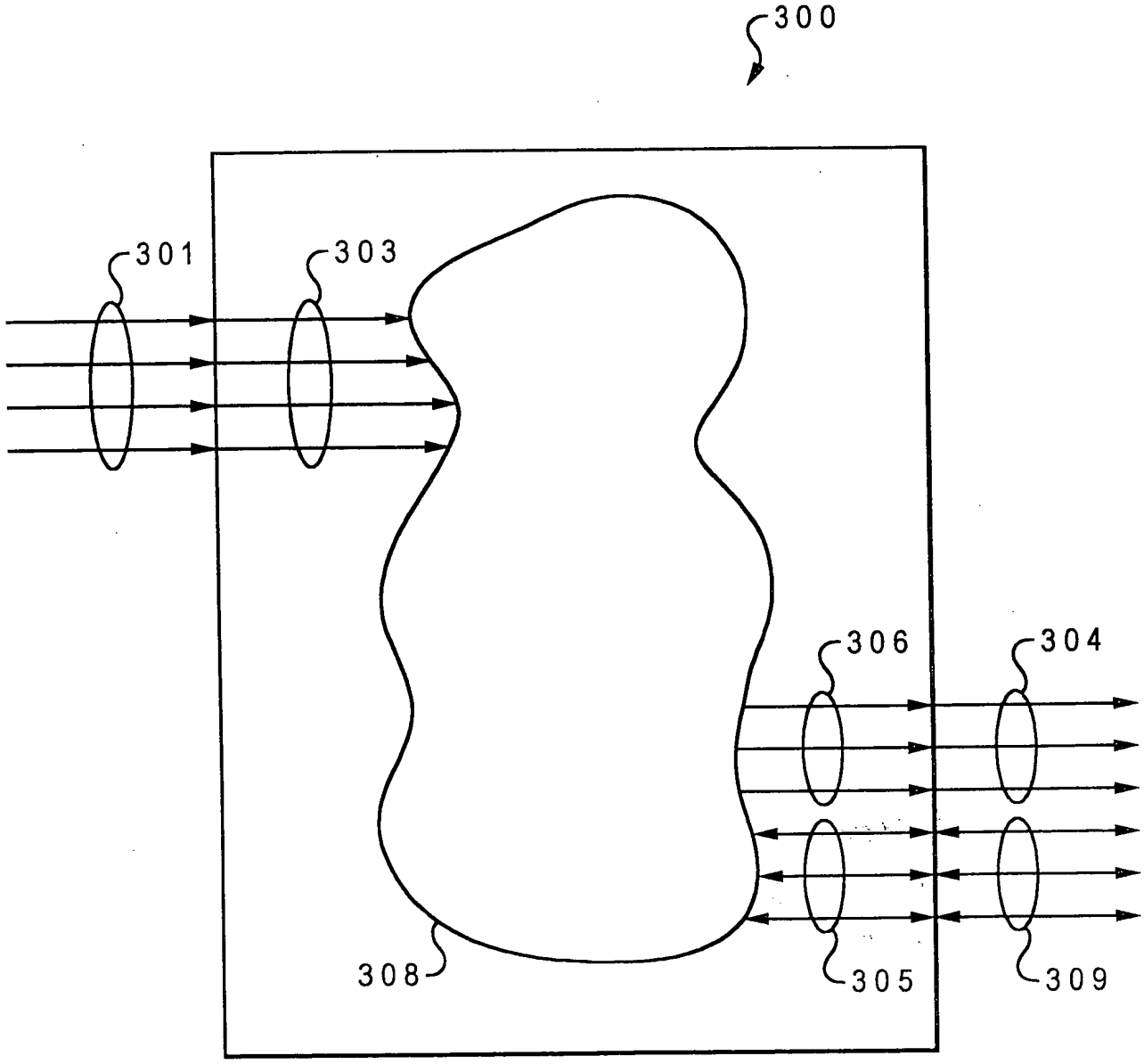


Fig. 3A

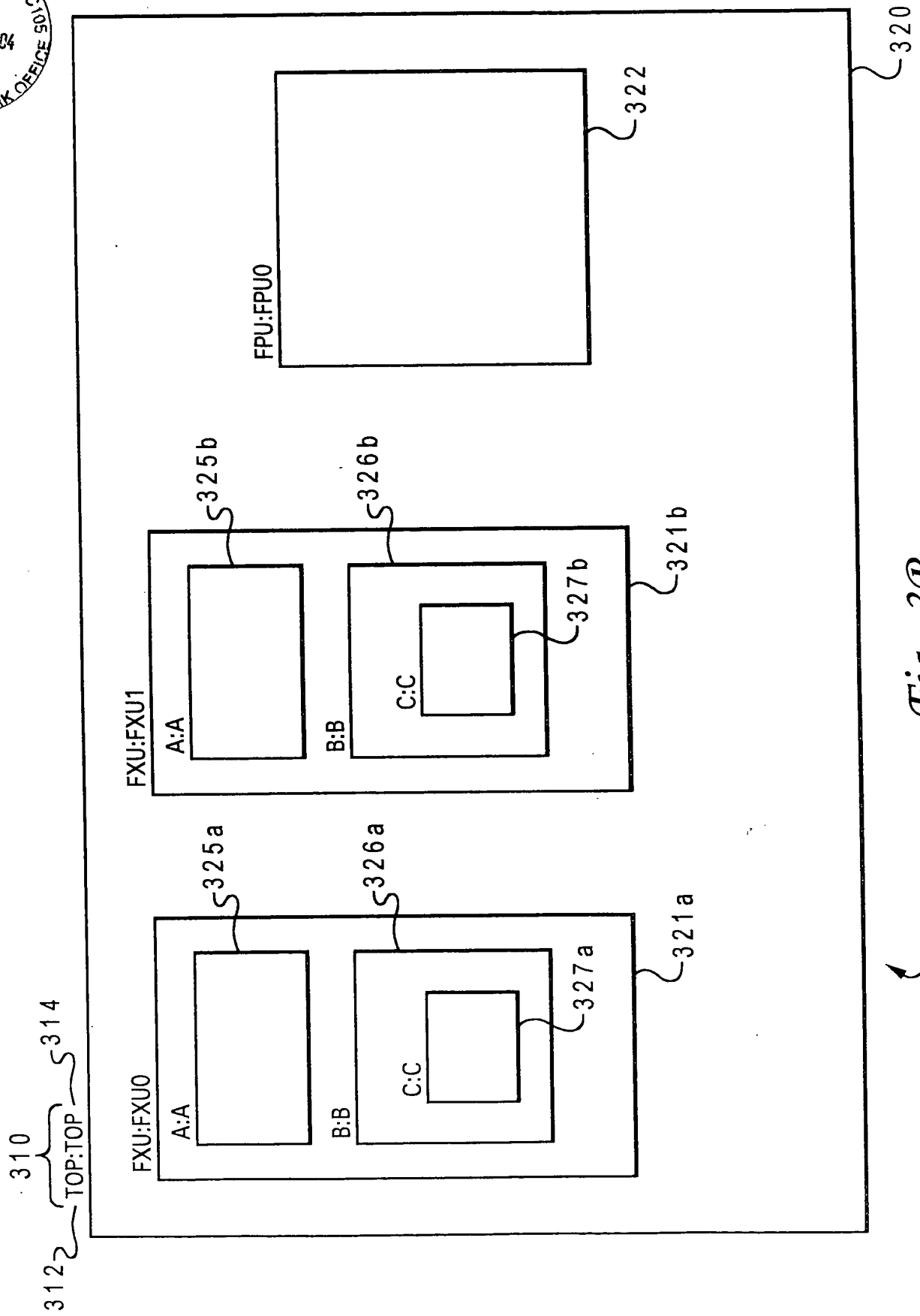


Fig. 3B

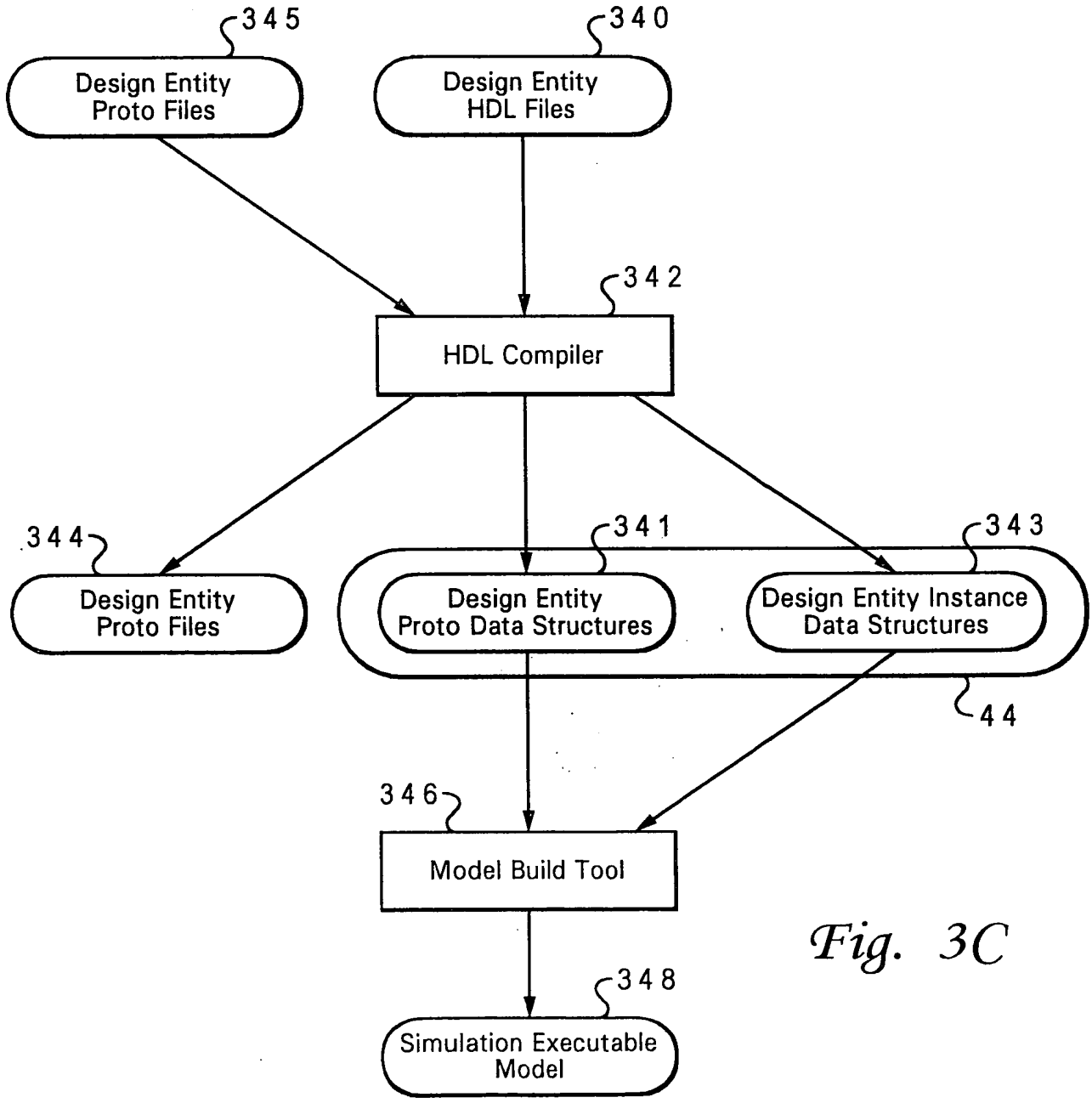


Fig. 3C

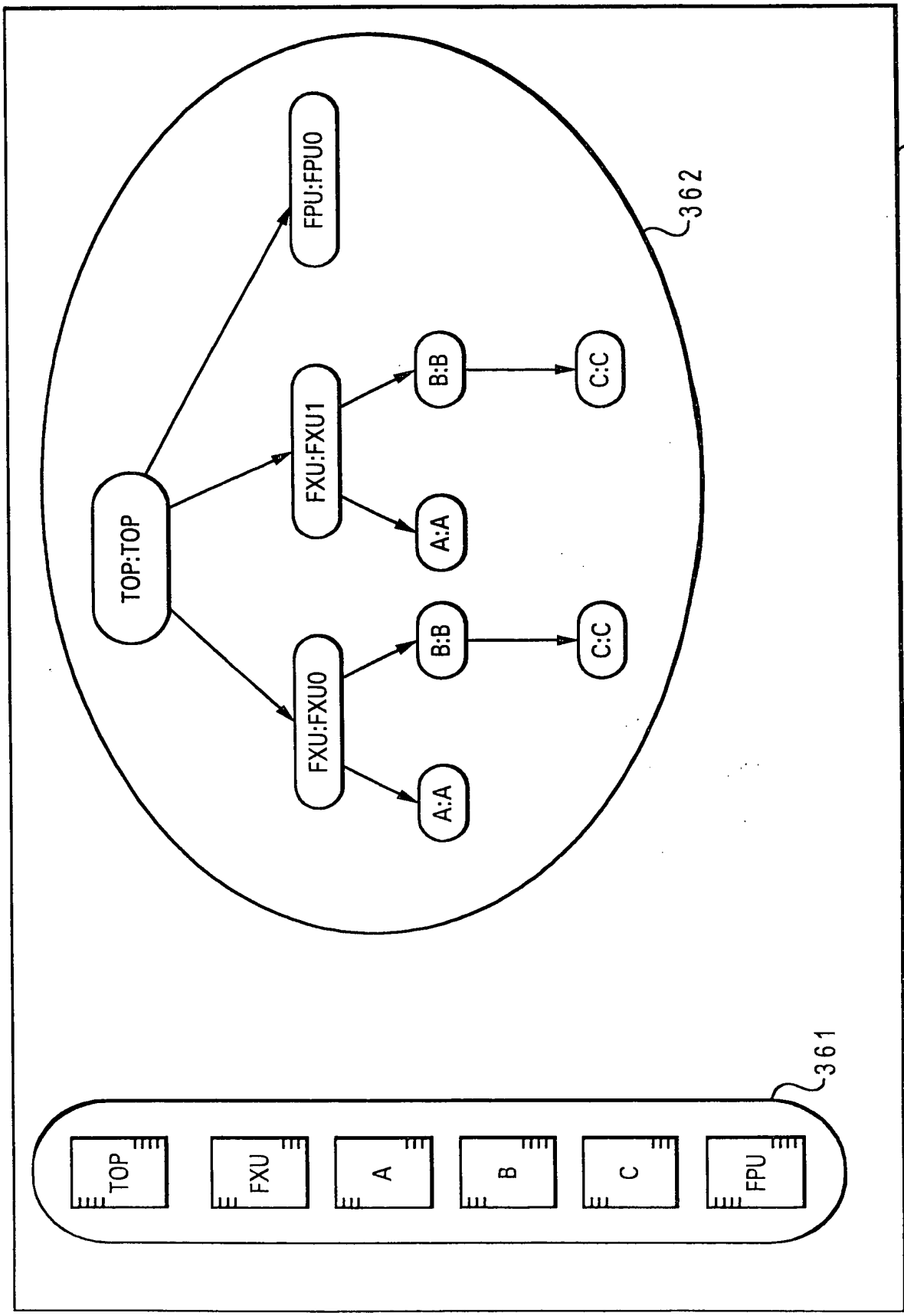


Fig. 3D

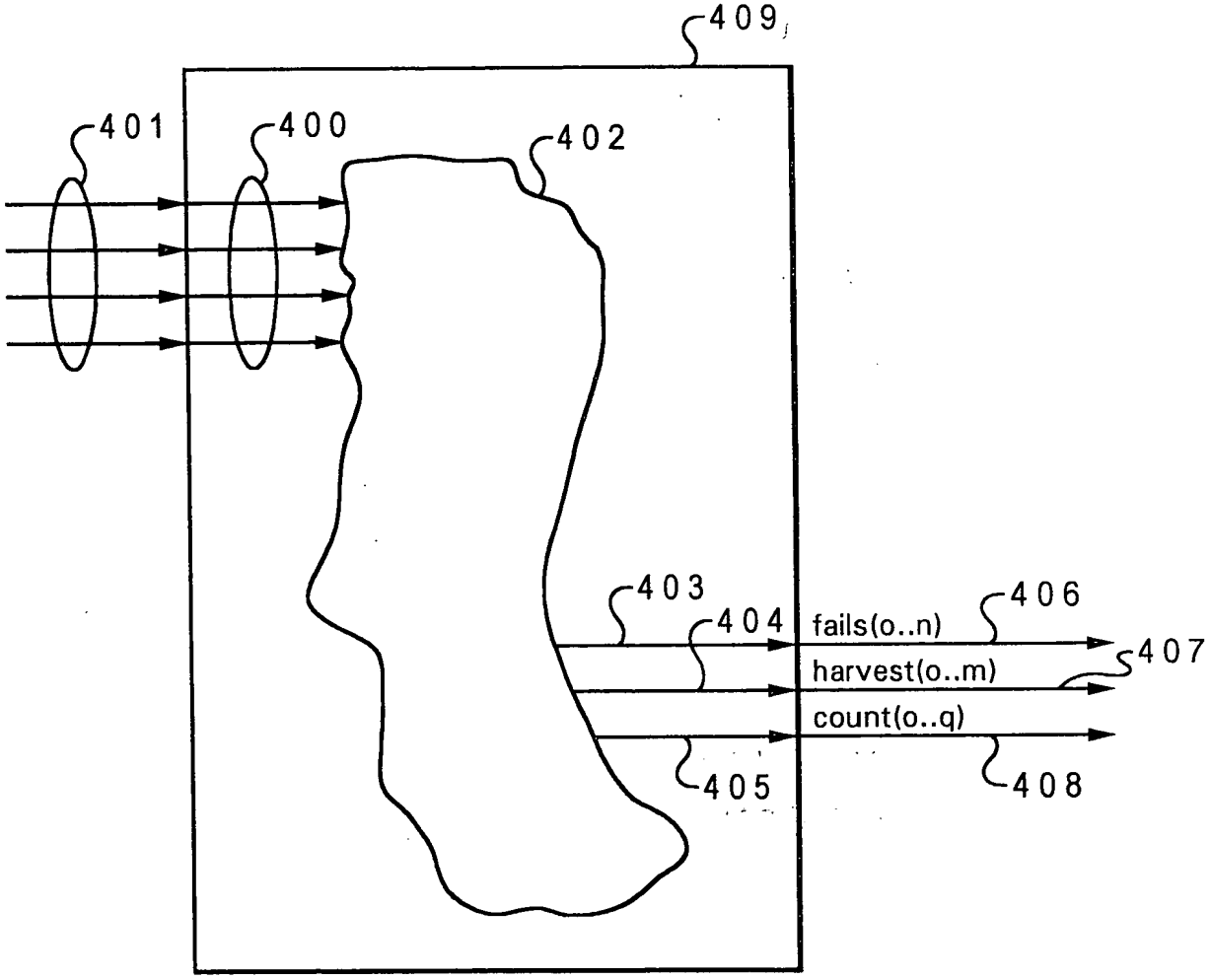


Fig. 4A

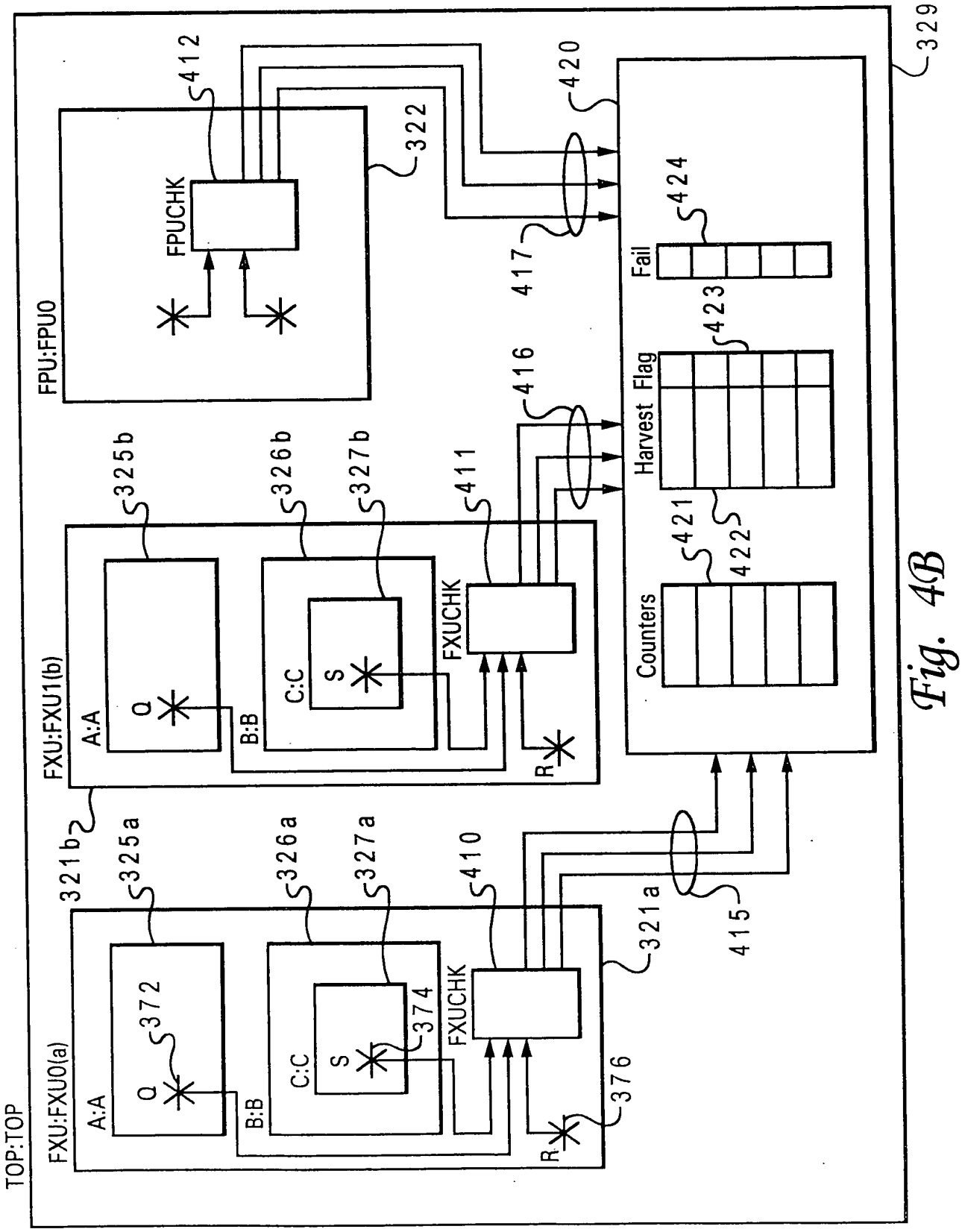


Fig. 4B

```
ENTITY FXUCHK IS

    PORT(   S_IN        :   IN std_ulogic;
            Q_IN        :   IN std_ulogic;
            R_IN        :   IN std_ulogic;
            clock       :   IN std_ulogic;
            fails       :   OUT std_ulogic_vector(0 to 1);
            counts      :   OUT std_ulogic_vector(0 to 2);
            harvests    :   OUT std_ulogic_vector(0 to 1);
        );

--!! BEGIN
--!! Design Entity: FXU;

--!! Inputs
--!! S_IN        =>    B.C.S;
--!! Q_IN        =>    A.Q;
--!! R_IN        =>    R;
--!! CLOCK       =>    clock;
--!! End Inputs

--!! Fail Outputs;
--!! 0 : "Fail message for failure event 0";
--!! 1 : "Fail message for failure event 1";
--!! End Fail Outputs;

--!! Count Outputs;
--!! 0 : <event0> clock;
--!! 1 : <event1> clock;
--!! 2 : <event2> clock;
--!! End Count Outputs;

--!! Harvest Outputs;
--!! 0 : "Message for harvest event 0";
--!! 1 : "Message for harvest event 1";
--!! End Harvest Outputs;

--!! End;

ARCHITECTURE example of FXUCHK IS

BEGIN

    ... HDL code for entity body section ...

END;
```

450 452 453 454 455 456 457 451 458 440

Fig. 4C

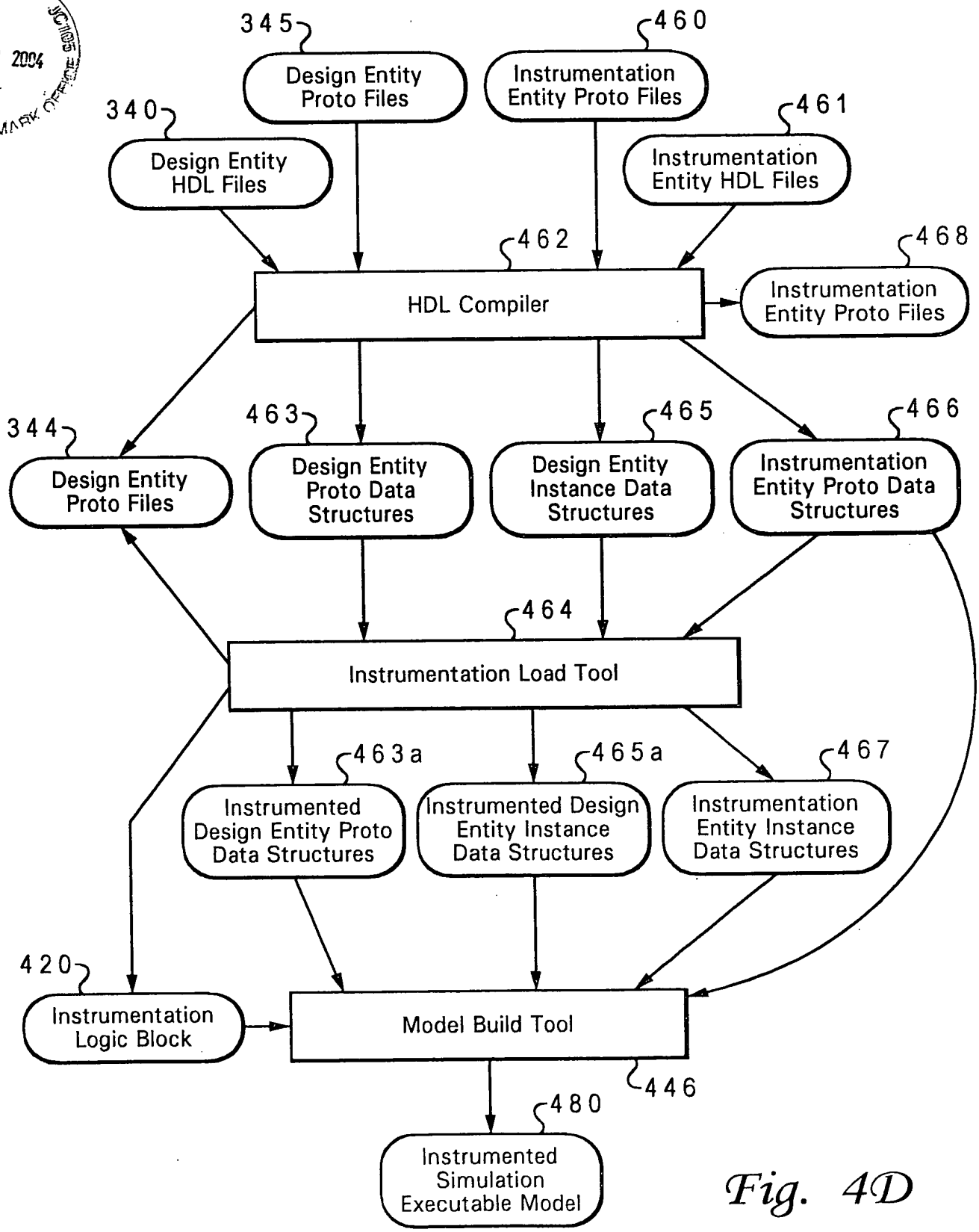


Fig. 4D

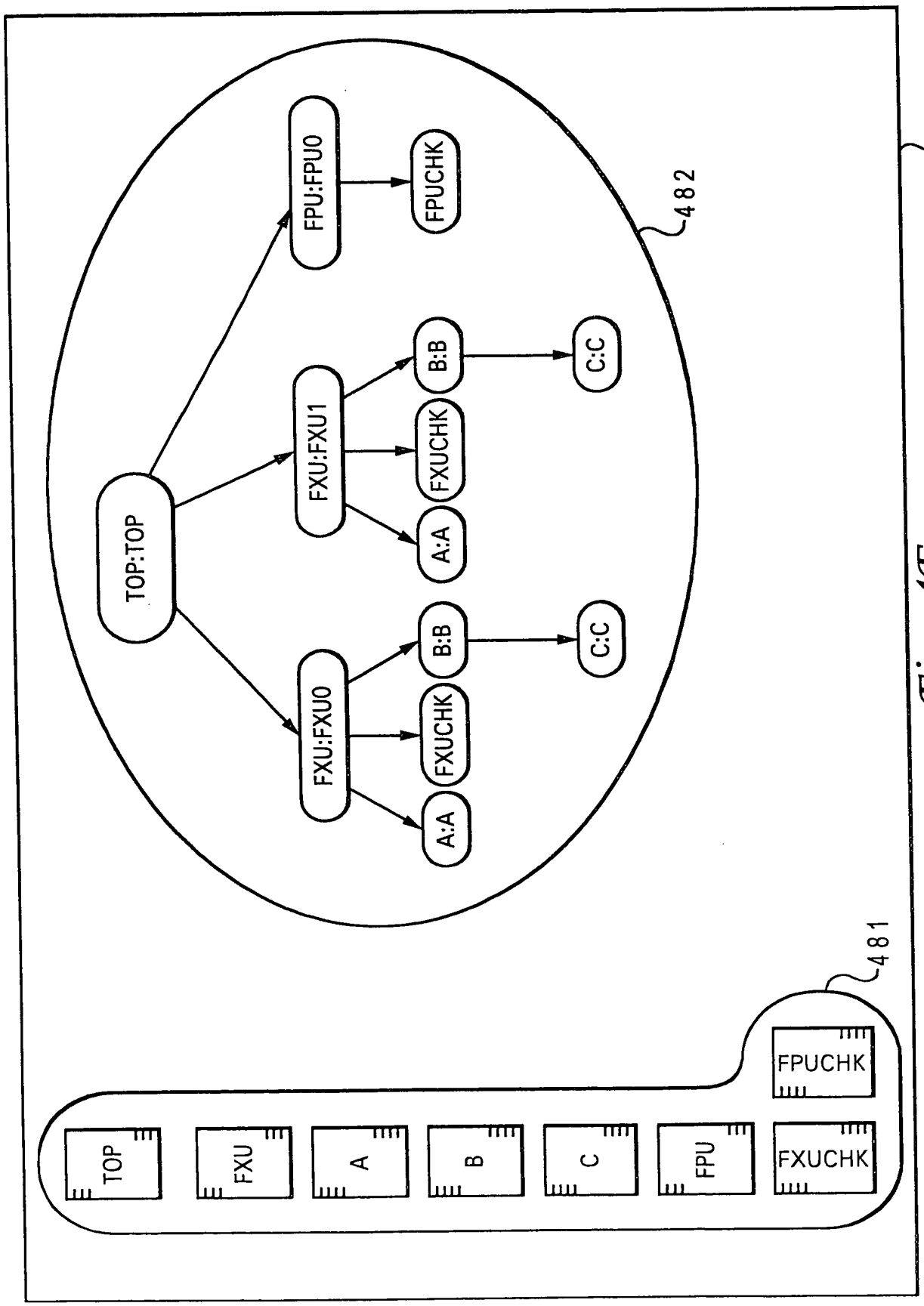


Fig. 4E

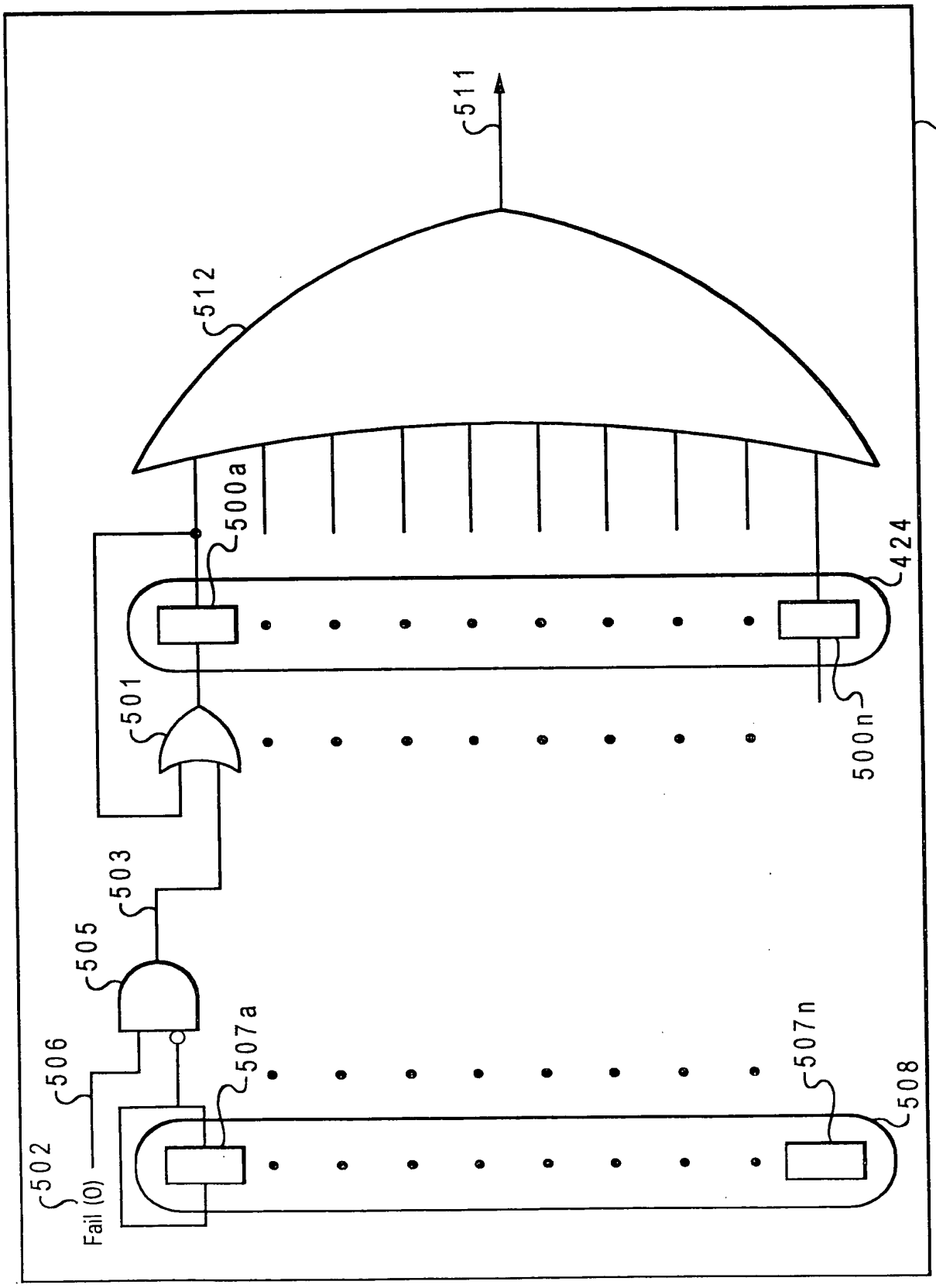


Fig. 5A

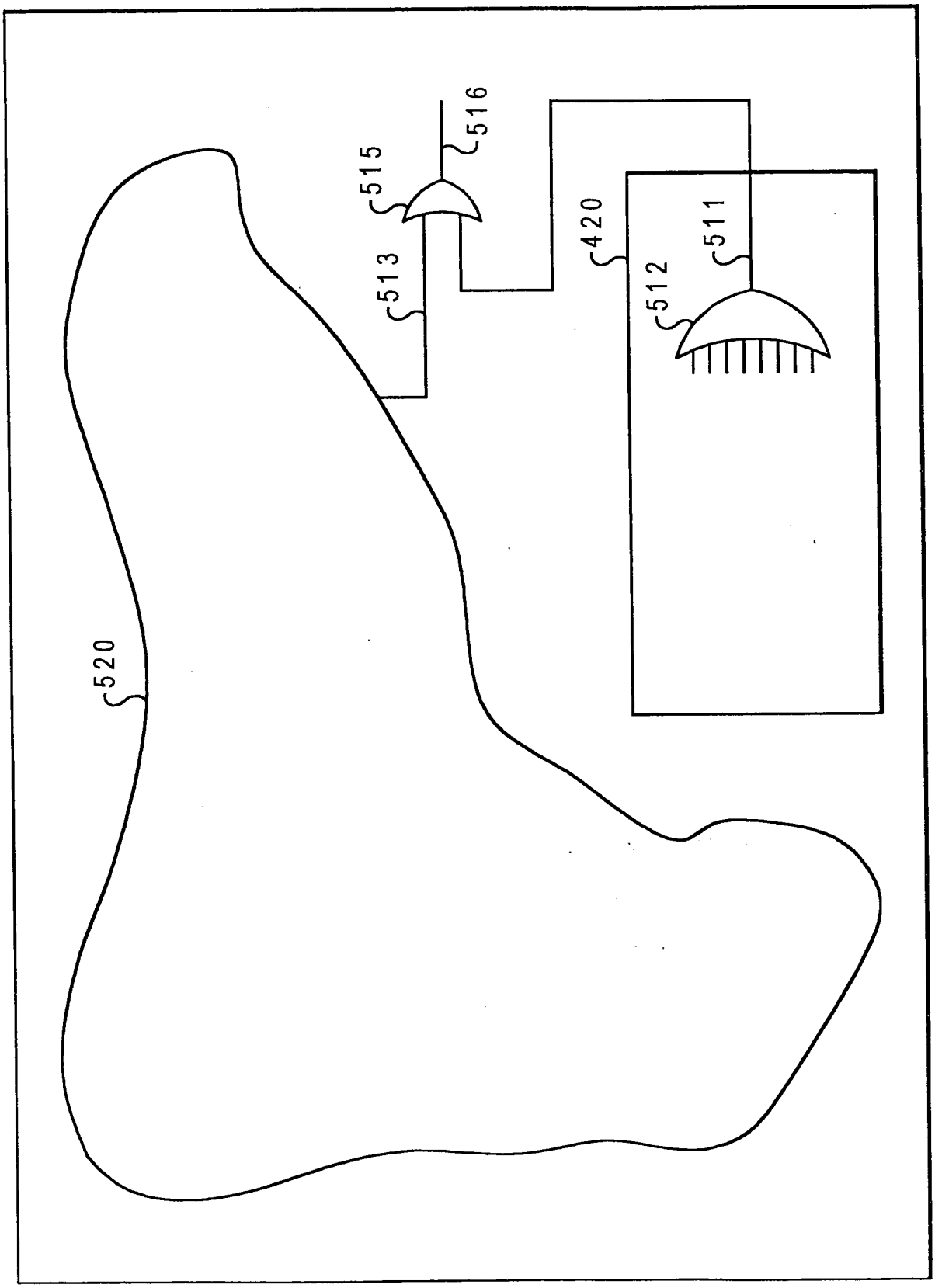


Fig. 5B

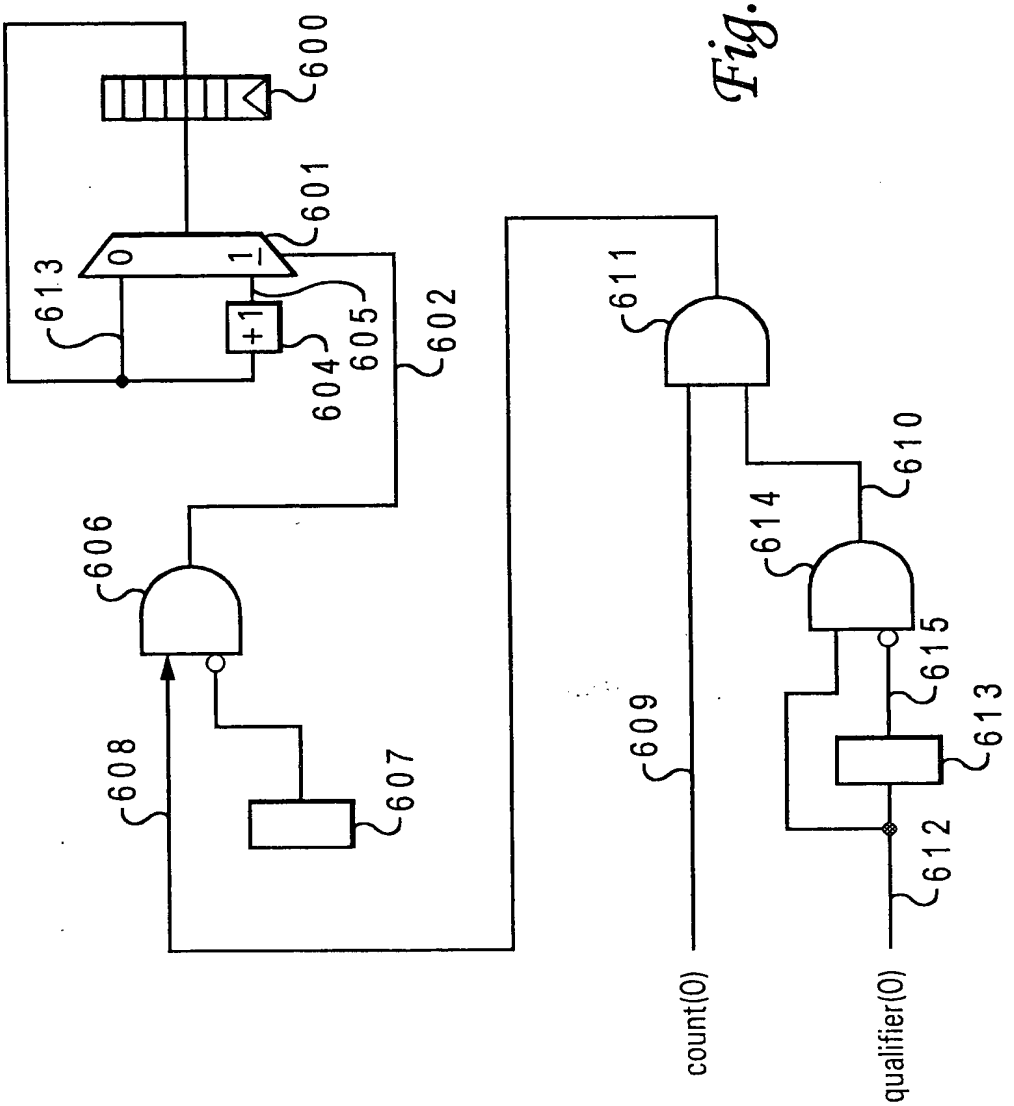


Fig. 6A

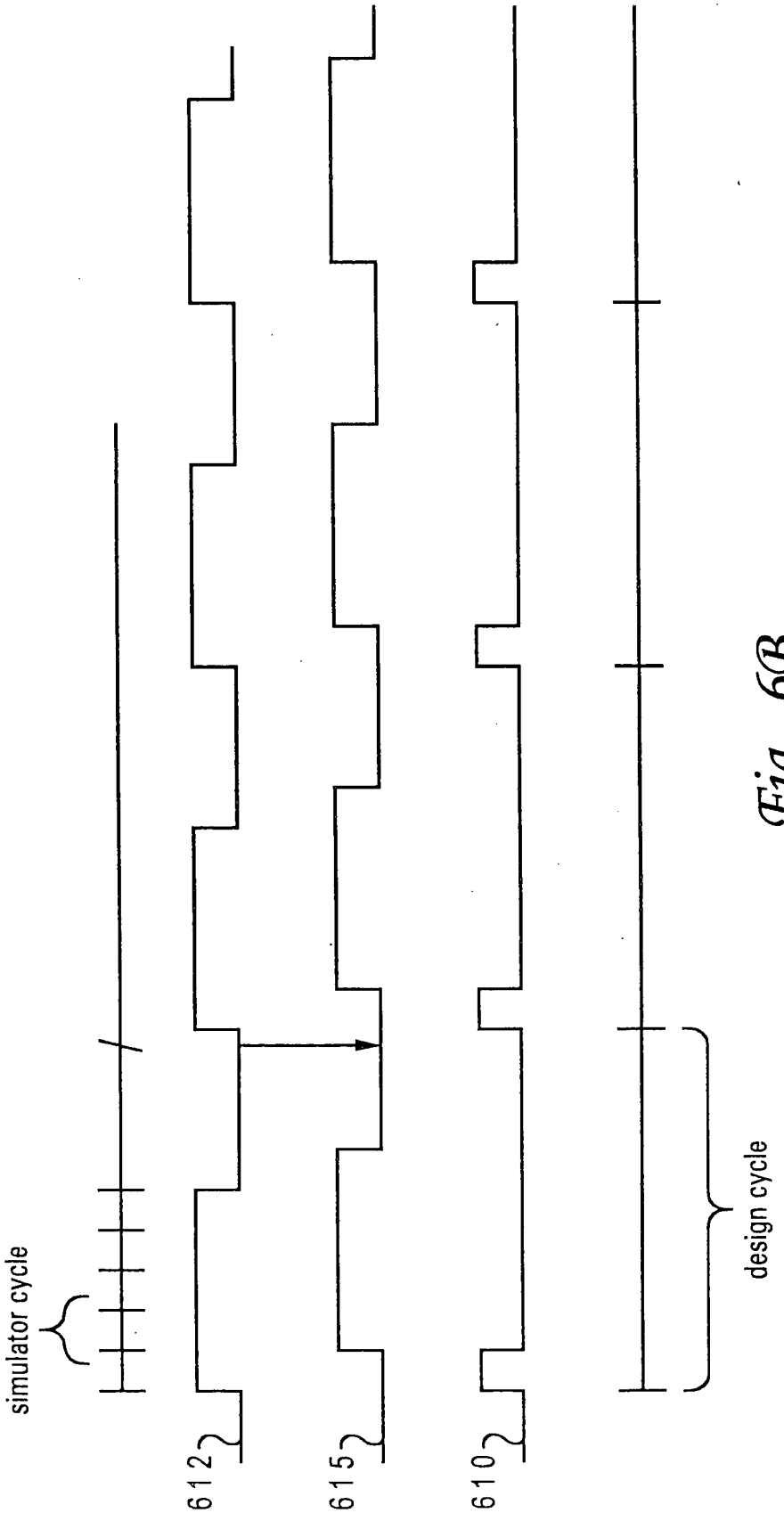


Fig. 6B

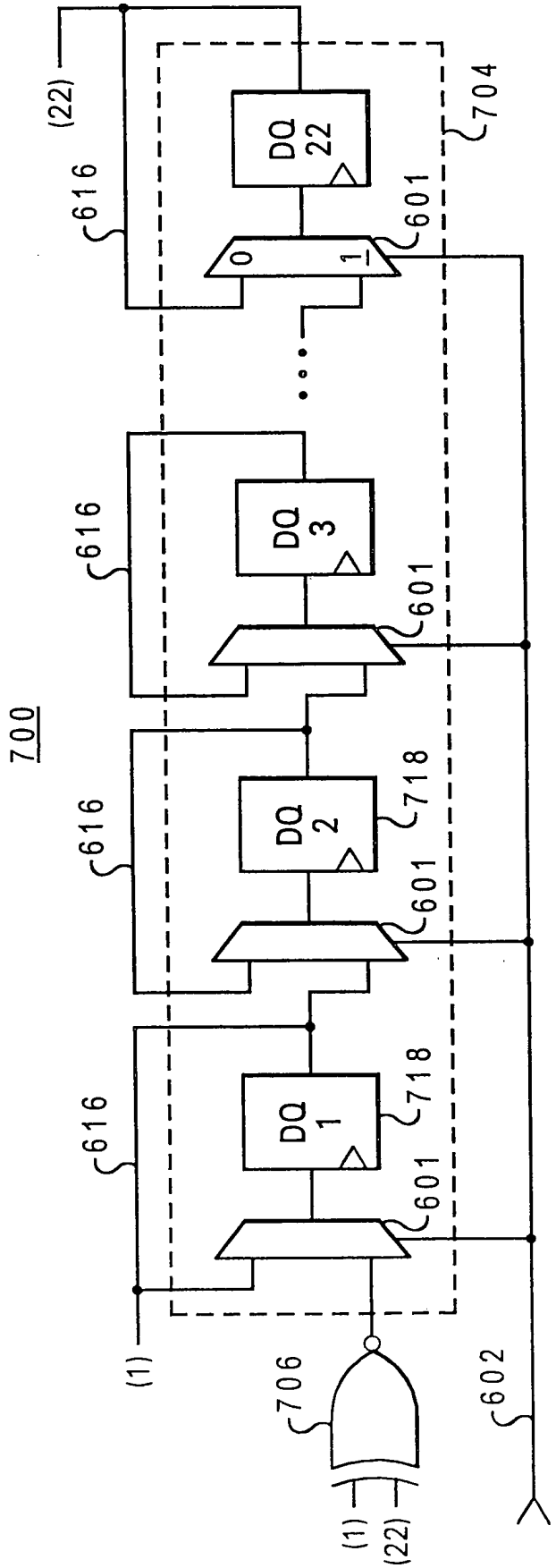


Fig. 7

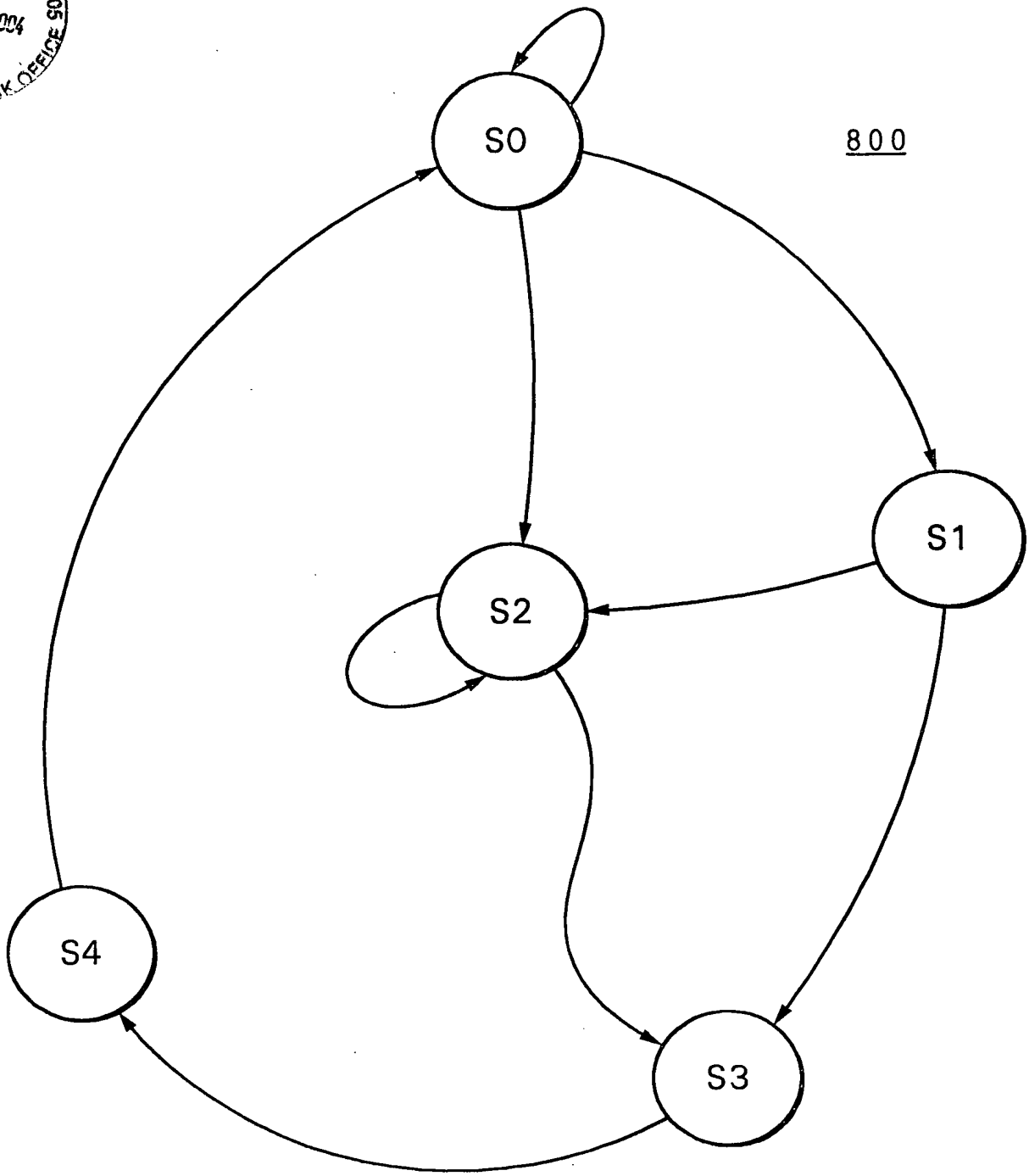


Fig. 8A
Prior Art

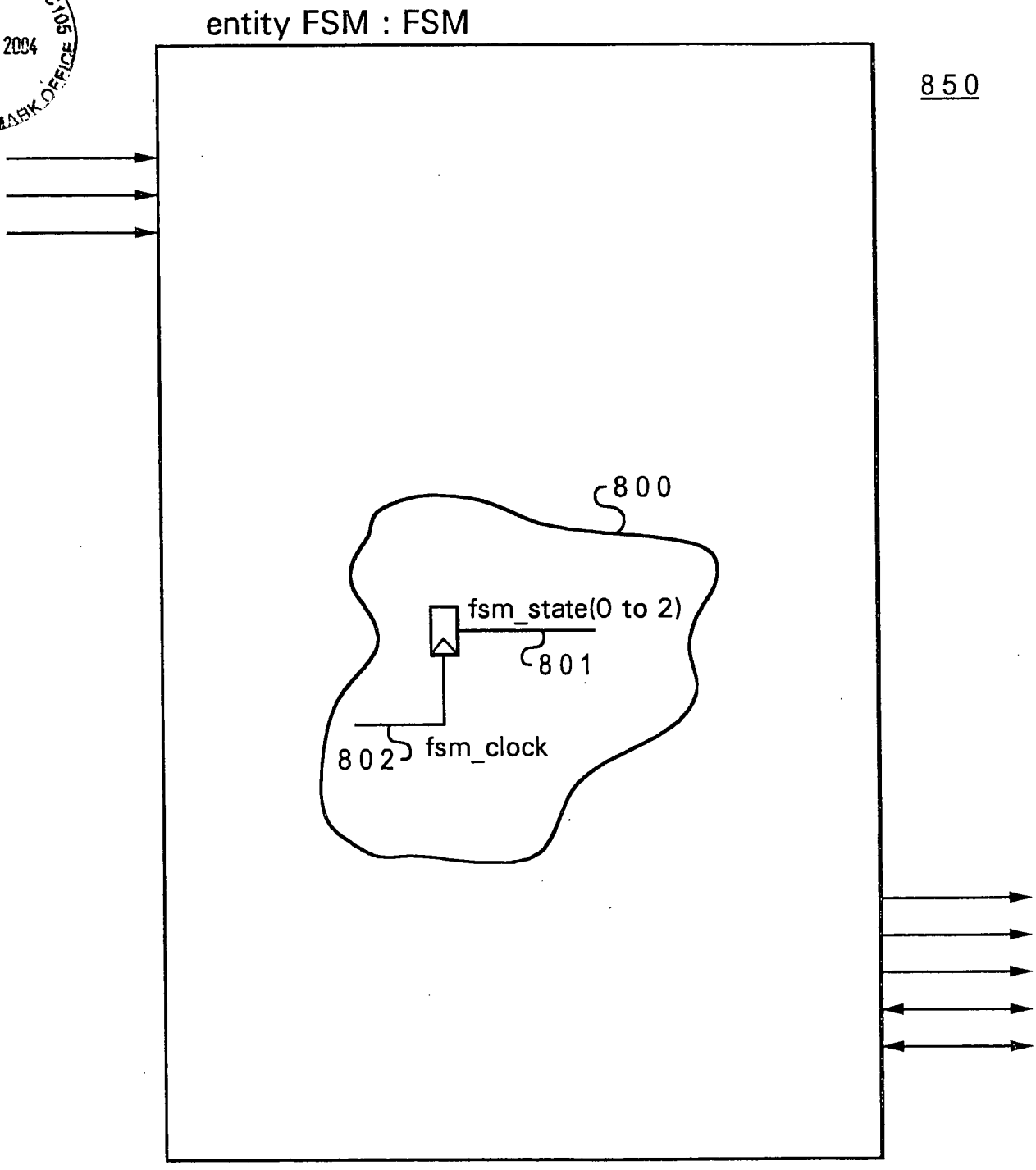


Fig. 8B
Prior Art

```
ENTITY FSM IS

    PORT(
            ....ports for entity fsm....
        );

    ARCHITECTURE FSM OF FSM IS

    BEGIN

            ... HDL code for FSM and rest of the entity ...

            fsm_state(0 to 2) <= ... Signal 801 ...

    853 --!! Embedded FSM : examplefsm;
    859 --!! clock        : (fsm_clock);
    854 --!! state_vector : (fsm_state(0 to 2));
    855 --!! states       : (S0, S1, S2, S3, S4);
    856 --!! state_encoding : ('000', '001', '010', '011', '100');
        --!! arcs         : (S0 => S0, S0 => S1, S0 => S2,
    857 --!!               (S1 => S2, S1 => S3, S2 => S2,
        --!!               (S2 => S3, S3 => S4, S4 => S0);
    858 --!! End FSM;

                ,
                ,
                ,
                ,
                ,
                ,
                ,

    END;
```

852 860

Fig. 8C

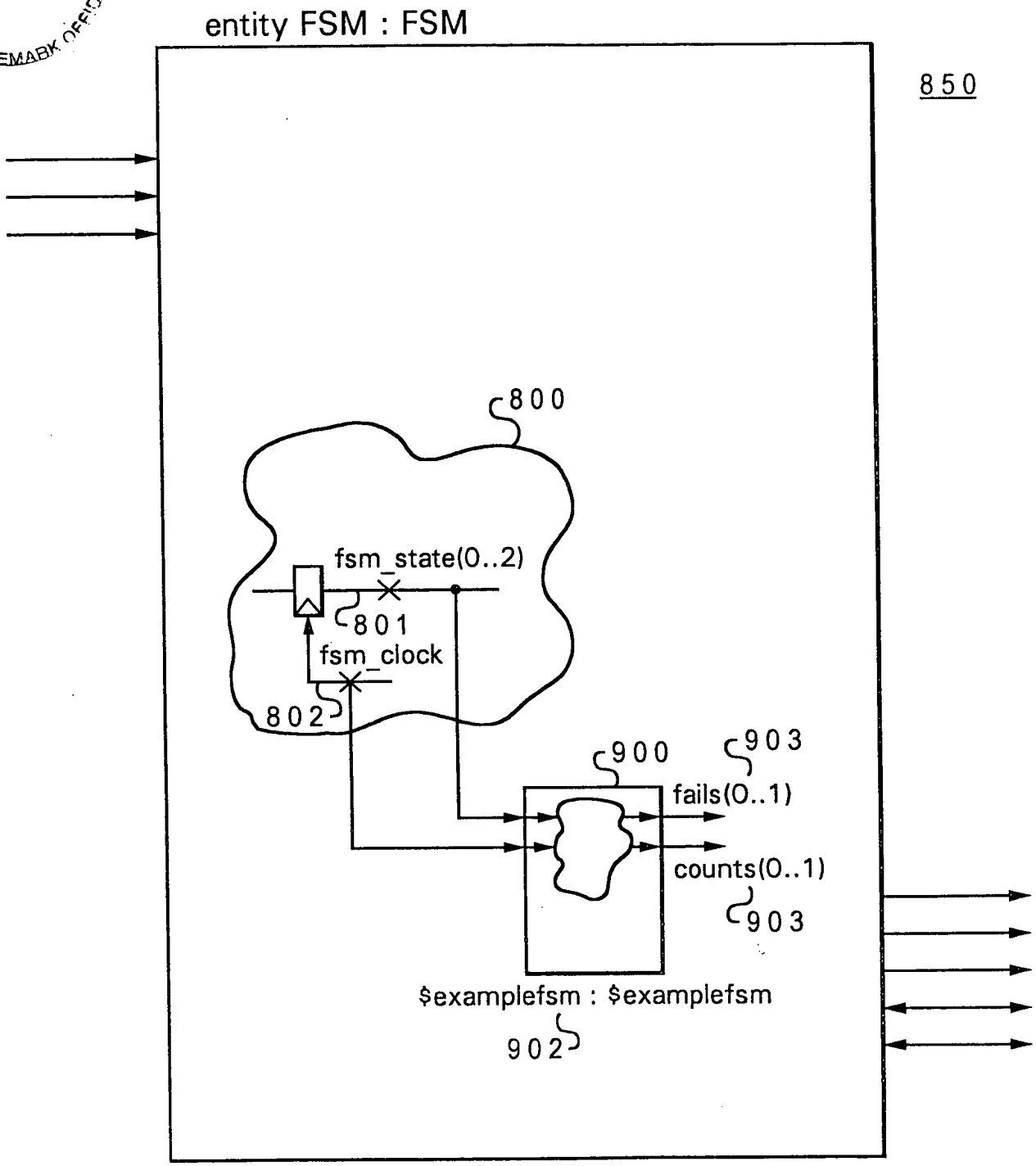
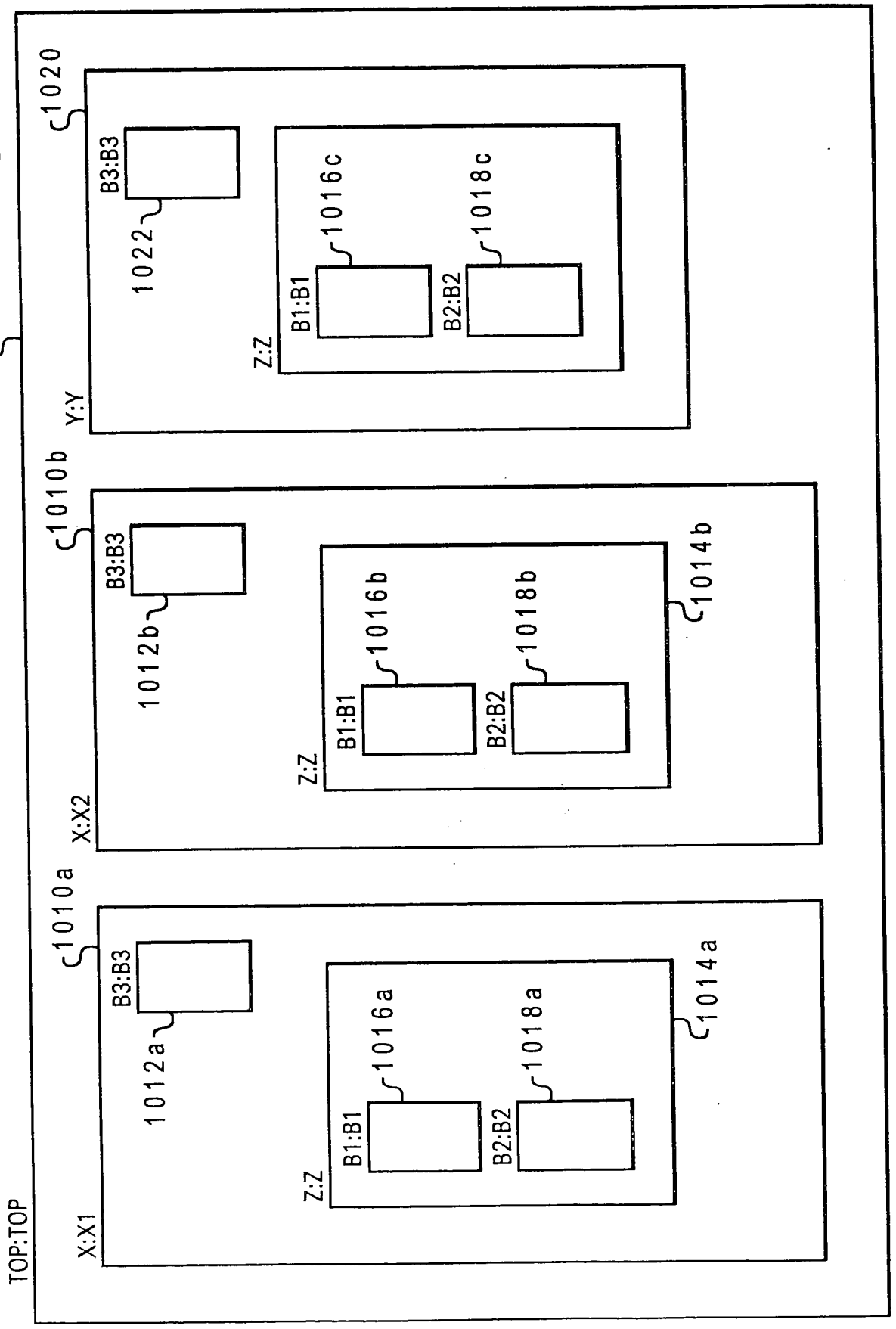


Fig. 9

Fig. 10A



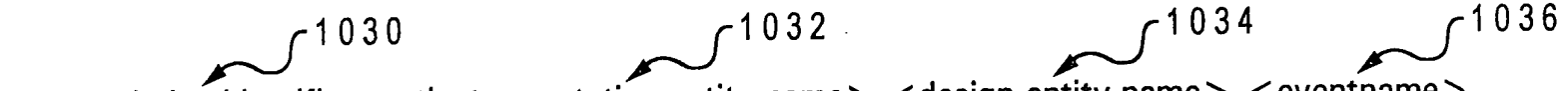


Fig. 10B

1030	1032	1034	1036	1040
X1	B3	X	COUNT1	1041
X1.Z	B1	Z	COUNT1	1042
X1.Z	B2	Z	COUNT1	1043
X2	B3	X	COUNT1	1044
X2.Z	B1	Z	COUNT1	1045
X2.Z	B2	Z	COUNT1	1046
Y	B4	Y	COUNT1	1047
Y.Z	B1	Z	COUNT1	1048
Y.Z	B2	Z	COUNT1	

Fig. 10C

Fig. 10D

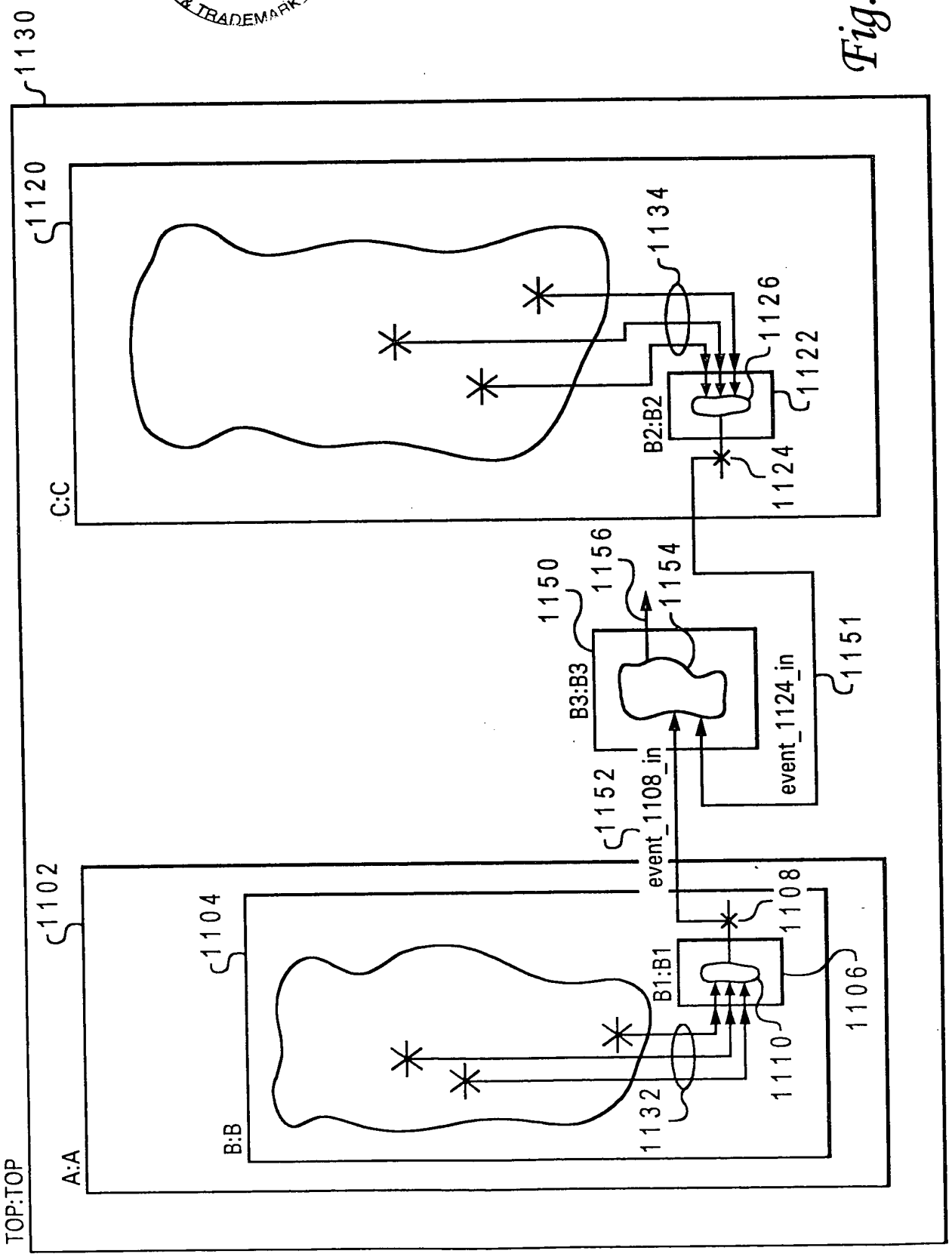


Fig. 11A

```
--!! Inputs                                                    1161
--!! event_1108_in <= C.[B2.count.event_1108];                 1162
--!! event_1124_in <= A.B.[B1.count.event_1124];
--!! End Inputs
```

1163 1164 1165 1166

Fig. 11B

```
--!! Inputs                                                    1171
--!! event_1108_in <= C.[count.event_1108];                    1172
--!! event_1124_in <= B.[count.event_1124];
--!! End Inputs
```

Fig. 11C

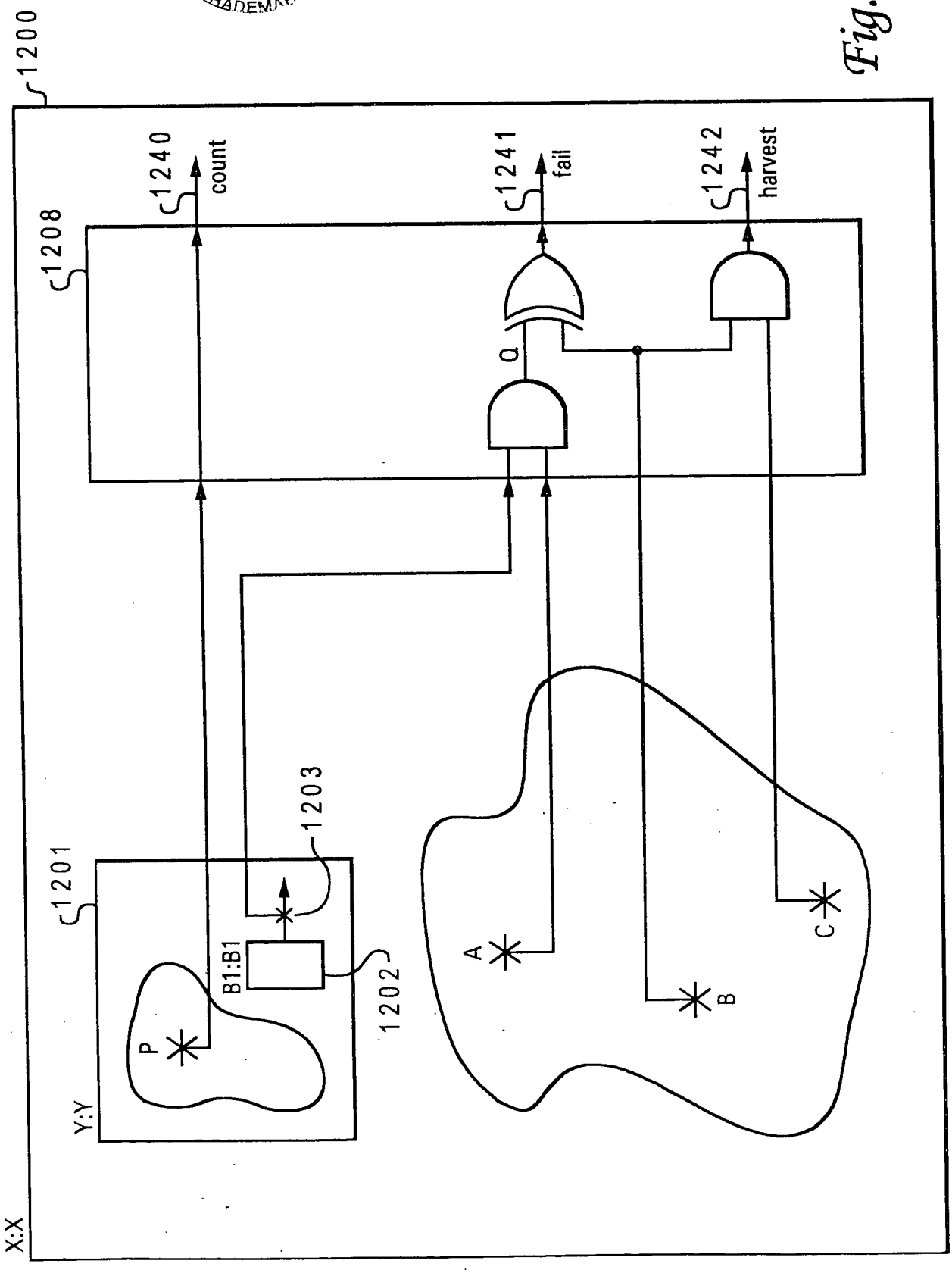


Fig. 12A

```
ENTITY X IS
    PORT(    :
             :
             :
         );
ARCHITECTURE example of X IS
BEGIN
      .
      .
      .
      .
      .
   ... HDL code for X ...
      .
      .
      .
      .
      .
   Y:Y
   PORT MAP(   :
               :
           );
   A <= ....
   B <= ....
   C <= ....
   --!! [count, countname0, clock] <= Y.P;
   --!! Q <= Y. [B1.count.count1] AND A;
   --!! [fail, failname0, "fail msg"] <= Q XOR B;
   --!! [harvest, harvestname0, "harvest msg"] <= B AND C;
END;
```

1220 1221 1222 1223 1230 1232 1234 1236

Fig. 12B